

Please amend the claims as follows. This listing of claims will replace all prior versions and Listings of Claims in the application:

Listing of Claims:

Claim 1 (Currently Amended): A method for creating a test summary report, comprising the operations of:

executing a computer software test application on a platform, the computer software test application having one or more test suites;

generating test results in an Extensible Markup Language (XML) enabled format; and processing using the XML enabled test results to create a test summary report, wherein the XML enabled test results is capable of being rearranged, the rearranged XML enabled test results including test suite tags, each test suite tag encapsulating the test results corresponding to each test suite of the computer software test application.

Claim 2 (Original): A method as recited in claim 1, wherein the test results are generated utilizing a status utility having functions that generate XML code.

Claim 3 (Original): A method as recited in claim 1, wherein the test results are output to a test execution log file, the test execution log file including a log of the test execution.

Claim 4 (Original): A method as recited in claim 3, further including the operation of processing the test execution log file to generate a well-formed XML test reports file.

Claim 5 (Original): A method as recited in claim 4, wherein the well-formed XML test reports file is further valid with respect to a Test document type definition (DTD).

Claim 6 (Currently Amended): A method as recited in claim 4, ~~further comprising the operation of logically arranging wherein a content of~~ the well-formed XML test reports file is reordered so as to create a logically arranged XML test reports file.

Claim 7 (Original): A method as recited in claim 6, wherein the logically arranged XML test reports file includes test suite tags indicating test reports that belong to particular test suites.

Claim 8 (Previously Presented): A method as recited in claim 6, wherein the test summary report is generated by converting the logically arranged XML test reports file into a Hypertext Markup Language (HTML) test summary report.

Claim 9 (Original): A method as recited in claim 8, wherein the HTML test summary report provides a test summary of the test execution log file.

Claim 10 (Original): A method as recited in claim 9, wherein the HTML test summary report includes links to failure description pages, wherein the failure description pages provide a detailed description of a particular test failure.

Claim 11 (Currently Amended): An Extensible Markup Language (XML) based report generator, comprising:

a parser that processes a test execution log file to generate a well-formed XML test reports file;

a logical parser ~~that processes operable to use~~ the well-formed XML test reports file to ~~produce reorder a content of the well-formed XML test reports file to produce~~ a logically arranged XML test reports file, the logically arranged XML test reports file including test suit

~~suite tags so as to indicate test reports belonging to particular test suits of a test application~~
encapsulating the test results corresponding to each test suite; and

an HTML converter parser that converts the logically arranged XML test reports file into an HTML test summary file.

Claim 12 (Previously Presented): An XML based report generator as recited in claim 11, wherein the well-formed XML test reports file is valid with respect to a Test document type definition (DTD).

Claim 13 (Cancelled)

Claim 14 (Previously Presented): An XML based report generator as recited in claim 11, wherein the HTML test summary report provides a test summary of the test execution log file.

Claim 15 (Previously Presented): An XML based report generator as recited in claim 14, wherein the HTML test summary report includes links to failure description pages, wherein the failure description pages provide a detailed description of a particular test failure.

Claim 16 (Currently Amended): A method for creating a test summary report, comprising the operations of:

executing a test application on a platform, wherein the test application is executed using a status utility having functions that generate XML code, the test application having one or more test suites;

generating test results in an Extensible Markup Language (XML) enabled format using the status utility, wherein the test results are output to a test execution log file;

processing using the test execution log file to generate a well-formed XML test reports file;

~~logically arranging~~ reordering a content of the well-formed XML test reports file to create a logically arranged XML test reports file wherein test suite tags are generated for each test suite, each test suite tag encapsulating the test results corresponding to each test suite; and

converting the logically arranged XML test reports file into an HTML test summary report.

Claim 17 (Original): A method as recited in claim 16, wherein the well-formed XML test reports file is further valid with respect to a Test document type definition (DTD).

Claim 18 (Cancelled)

Claim 19 (Original): A method as recited in claim 16, wherein the HTML test summary report provides a test summary of the test execution log file.

Claim 20 (Original): A method as recited in claim 19, wherein the HTML test summary report includes links to failure description pages, wherein the failure description pages provide a detailed description of a particular test failure.

Claim 21 (New): A method of generating and processing test results, comprising the operations of:

executing one or more tests from a plurality of test suites and generating test execution log file wherein each test result includes data identifying the test to which the test result relates and the test suite to which the test belongs;

processing the test execution log file to generate a well-formed XML based test reports file wherein the test results in the test reports file are arranged as a plurality of independent test results, each including a test ID and data identifying a test suite to which a test belongs;

processing the well-formed XML test reports file to logically arrange the well-formed XML test reports file to create a logically arranged XML test reports file in which tags are generated for each test suite the tags encapsulating the test results corresponding to each test suite; and

outputting the logically arranged well-formed XML test reports file.